Exemplo 2:

```
início
  inteiro: N, FAT;
  caracter: TÍTULO;
  procedimento FATORIAL (NF, FATF);
    inteiro: NF, FATF;
    início
      inteiro: I;
      FATF ← 1;
      para I de 1 até NF faça
        FATF ← FATF * I;
      fim para;
      imprima (TÍTULO, NF, "É", FATF);
    fim; [FATORIAL]
  TÍTULO ← "O FATORIAL DE";
  N ← 5;
  FATORIAL (N, FAT);
  imprima (FAT, "E", TÍTULO, N);
fim.
```

Será impresso:

```
O FATORIAL DE 5 é 120
120 É O FATORIAL DE 5
```

Um procedimento pode chamar outro procedimento, este segundo chamar um terceiro procedimento, e assim por diante. Diz-se que os procedimentos estão aninhados.

Exemplo 3:

```
início
  inteiro: PÁGINA;
  procedimento LINHA (N);
    inteiro: N;
    início
      procedimento CABEC;
        início
          imprima ("EXEMPLO", "PAG.", PÁGINA);
          PÁGINA ← PÁGINA + 1;
        fim; [CABEC]
      se N mod 60 = 0
        então
          CABEC;
      fim se;
      imprima ("LINHA", N);
    fim; [LINHA]
  inteiro: I;
  PÁGINA ← 1;
  para I de 0 até 1200 faça
    LINHA (I);
  fim para;
fim.
```


Será produzida a saída:

```
EXEMPLO                         PÁG. 1
      LINHA 0
      LINHA 1
      LINHA 2
        .
        .
        .
      LINHA 59

EXEMPLO                         PÁG. 2
      LINHA 60
      LINHA 61
        .
        .
        .
      LINHA 119

EXEMPLO                         PÁG. 3
      LINHA 120
        .
        .
        .
```

7.4 FUNÇÃO

Quando se necessitar atribuir o resultado da chamada de um procedimento a uma variável, utilizar este resultado numa expressão aritmética ou imprimir este resultado, é necessário que exista um parâmetro de retorno na chamada do procedimento e este parâmetro é que será utilizado.

Por exemplo:

```
ABS (– 3, Y);
X ← Y * 2;
```

Seria mais conveniente se pudéssemos escrever:

```
X ← ABS (– 3) * 2;
```

Isto é possível utilizando um tipo especial de procedimento, denominado função cuja sintaxe é a seguinte: